EVENTCONE-WM: COUNTERFACTUAL EVENT CONES FOR RELIABLE ACTION-CONDITIONED LATENT PLANNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Action-conditioned world models can predict plausible futures while assigning an action's effects to entities that it cannot physically reach. We introduce EventCone-WM, which contrasts candidate and matched-control latent rollouts, decodes typed entity–event relations, and propagates them into a finite-horizon *event cone*. Outside-cone invariance suppresses unsupported changes, inside-cone separation preserves task-relevant effects, and paired-action ranking connects localized prediction to planning. We evaluate six methods across three manipulation environments, four evaluation conditions, and five seeds, totaling 36,000 closed-loop episodes. EventCone-WM reaches 76.8% mean OOD success versus 61.4% for JEPA-WM, a 15.4-point gain. Across all conditions, it reduces cone false-positive rate from 24.5% to 11.1% and collision rate from 14.9% to 8.7%, while increasing per-action latency from 68.4 to 83.6 ms. Component, corruption, entity-scaling, sensitivity, and failure analyses consistently link localized action effects to improved closed-loop decisions.

1 INTRODUCTION

World models enable simulation (Ha & Schmidhuber, 2018), pixel-based planning (Hafner et al., 2019), latent imagination (Hafner et al., 2020a;b; 2025), and continuous-control planning (Hansen et al., 2024). Manipulation also requires localized effects: a pushed block may hit a cup, but a distant lamp should remain unchanged. We hypothesize that unsupported changes can mislead action selection.

Visual self-supervision supplies reusable features (Caron et al., 2021; Oquab et al., 2023). Joint-embedding prediction (LeCun, 2022) extends to images (Assran et al., 2023) and video (Bardes et al., 2024); DINO-WM (Zhou et al., 2025), V-JEPA 2 (Assran et al., 2025), and JEPA-WM (Terver et al., 2026) connect feature prediction to physical planning. We seek an additional constraint: predicted action effects should follow reachable interactions.

Object–relation dynamics (Battaglia et al., 2016; Kipf et al., 2020) and object discovery or tracking (Greff et al., 2019; Locatello et al., 2020; Kipf et al., 2022) provide structural interfaces. Causal-JEPA uses object-level latent masking for interaction-dependent prediction (Nam et al., 2026); our proposal instead couples candidate/control rollouts with finite-horizon event reachability.

Causal inference (Pearl, 2009) and causal representation learning (Schölkopf et al., 2021) distinguish mechanisms from association; counterfactual policy search evaluates alternative actions (Buesing et al., 2019). Our paired-input interpretation does not identify a full structural causal model. Following uncertainty-aware planning (Chua et al., 2018), we retain separate uncertainty penalties: cone sparsity is not a safety guarantee.

Meta-World (Yu et al., 2020), ManiSkill3 (Tao et al., 2024), and RoboCasa (Nasiriany et al., 2024) motivate manipulation evaluation; CausalWorld (Ahmed et al., 2021) motivates controlled shifts; IntPhys 2 (Bordes et al., 2025) supplies physical-consistency diagnostics; and Diffusion Policy (Chi et al., 2023) offers complementary visuomotor control. We focus on Push-T, Meta-World Push, and Meta-World PickPlace because they expose distinct contact structures while retaining controlled access to event annotations.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

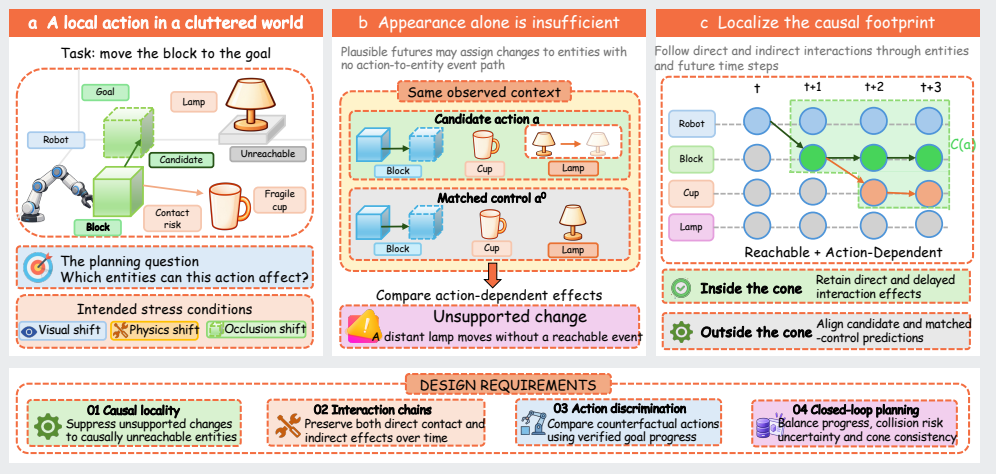

Figure 1: Problem setting and requirements. (a) Local actions propagate through direct or delayed interactions. (b) Candidate/control pairs expose unsupported changes. (c) Event reachability defines the entity–time footprint used for inside/outside constraints, action discrimination, and planning.

We propose EventCone-WM for *counterfactual localization*. As Figure 1 illustrates, paired candidate/control rollouts expose unsupported changes, while event paths retain direct and delayed effects along interaction chains. Their finite entity–time cone supervises outside invariance, inside-effect separation, event prediction, action ranking, and cone-aware planning.

The methodological contributions are:

- finite-horizon event cones from paired action-conditioned latent rollouts;
- event supervision, outside invariance, inside effect separation, and action-pair ranking;
- cone-aware planning costs for collisions, uncertainty, and unsupported change; and
- an evaluation design that links closed-loop performance to object-level mechanism diagnostics.

2 Related Work

World models and latent prediction. World models compress observations into predictive states that support simulation and control (Ha & Schmidhuber, 2018). Rather than reconstructing every pixel, joint-embedding predictive architectures learn to predict representations of missing or future observations (LeCun, 2022). DINO-WM shows that frozen self-supervised visual features can support action-conditioned prediction and planning (Zhou et al., 2025). V-JEPA 2 scales self-supervised video pretraining and adds an action-conditioned model for planning (Assran et al., 2025). JEPA-WM systematically studies representation, predictor, and planning choices (Terver et al., 2026). Our proposal uses this latent-prediction family as its backbone but adds a structured constraint on where an action-induced change may occur.

Events, objects, and causal structure. Object-centric and graph-based dynamics models represent interactions more explicitly than monolithic frame predictors. Causal-JEPA uses object-level latent masking to encourage interaction-dependent prediction (Nam et al., 2026). Our use of "causal" is operational and deliberately narrower: the event cone is identified from paired interventional inputs and predicted relation reachability within a finite horizon. It is not a claim of recovering the full causal graph of the environment from passive video.

Embodied evaluation. Meta-World supplies standardized robotic manipulation tasks (Yu et al., 2020); ManiSkill3 emphasizes scalable simulation and manipulation evaluation (Tao et al., 2024);

and RoboCasa provides diverse household manipulation scenes (Nasiriany et al., 2024). Push-T and the Meta-World Push and PickPlace tasks form our controlled evaluation suite. IntPhys 2 offers complementary diagnostics for intuitive physical understanding (Bordes et al., 2025), but it lacks the closed-loop action interface required for the primary claim.

3 Problem Formulation

Let an observation o_t contain N entities and let a_t be a continuous robot action. An encoder produces entity tokens

$$Z_t = f_\theta(o_{t-K+1:t}) = \{z_t^1, \dots, z_t^N\}, \qquad z_t^i \in \mathbb{R}^d. \tag{1}$$

An action-conditioned predictor rolls the tokens forward for horizon H,

$$\widehat{Z}^a_{t+1:t+H} = F_\phi(Z_t, a_{t:t+H-1}). \tag{2}$$

For every candidate sequence a, we construct a matched control sequence a^0. In the simplest case a^0 is the zero action; for a moving camera or nonstationary robot it preserves the same camera and proprioceptive evolution while removing task-space intervention. The paired latent difference for entity i and step h is

$$d_{i,h}(a, a^0) = \left\| \widehat{z}^{i,a}_{t+h} - \mathrm{sg}(\widehat{z}^{i,a^0}_{t+h}) \right\|_2, \tag{3}$$

where sg stops gradients through the reference branch.

An event decoder g_ψ predicts a directed, typed relation graph $G_{t+h} = (V, E_{t+h})$. Relations are multi-label: contact, support, containment, and robot manipulation may co-occur on an ordered pair. If p_{ijhk} is the sigmoid probability of type k, the probability of any propagating edge is

$$e_{ijh} = 1 - \prod_{k \in \mathcal{K}_{\mathrm{prop}}} (1 - p_{ijhk}). \tag{4}$$

Soft reachability starts at the robot node, $q_{r,0} = 1$ and $q_{i \neq r,0} = 0$, and applies a noisy-OR recurrence,

$$q_{j,h} = 1 - (1 - q_{j,h-1}) \prod_i (1 - q_{i,h-1} e_{ijh}). \tag{5}$$

The effect gate $u_{i,h} = \sigma((\mathrm{sg}(d_{i,h}) - \tau_d)/T_d)$ is detached from the latent-distance gradient. Soft membership is $c_{i,h} = q_{i,h} u_{i,h}$, whereas evaluation uses the hard set

$$\mathcal{C}(a) = \{(i,h) : q_{i,h} \geq \tau_q,\ d_{i,h} > \tau_d\}. \tag{6}$$

Training masks use detached structural reachability $\widetilde{c}_{i,h} = \mathrm{sg}(q_{i,h})$, not the effect gate; reducing $d_{i,h}$ therefore cannot directly remove an entity from its own supervision. Appendix A gives the recurrence, empty-cone convention, and gradients.

4 Method

Figure 2 connects paired latent prediction and typed events to cone construction, multi-objective training, and receding-horizon control; the same cone supplies representation constraints and planner costs.

4.1 Action-conditioned latent backbone

The implementation reuses a frozen visual encoder and a transformer predictor, following the practical JEPA-WM setup (Terver et al., 2026). Entity tokens are obtained from environment segmentation for the primary comparison and from tracked proposals in the observation ablation. Type, 2-D position, depth, and proprioceptive embeddings are added to each token. The predictor alternates temporal self-attention with relation-biased entity attention, allowing the method to attach to an existing backbone instead of requiring a new video foundation model.

The base future-prediction loss is

$$\mathcal{L}_{\mathrm{pred}} = \frac{1}{NH} \sum_{h=1}^{H} \sum_{i=1}^{N} \left\| \widehat{z}^{i,a}_{t+h} - \mathrm{sg}(\bar{z}^i_{t+h}) \right\|_1, \tag{7}$$

where $\bar{z}$ is produced by an exponential-moving-average target encoder.

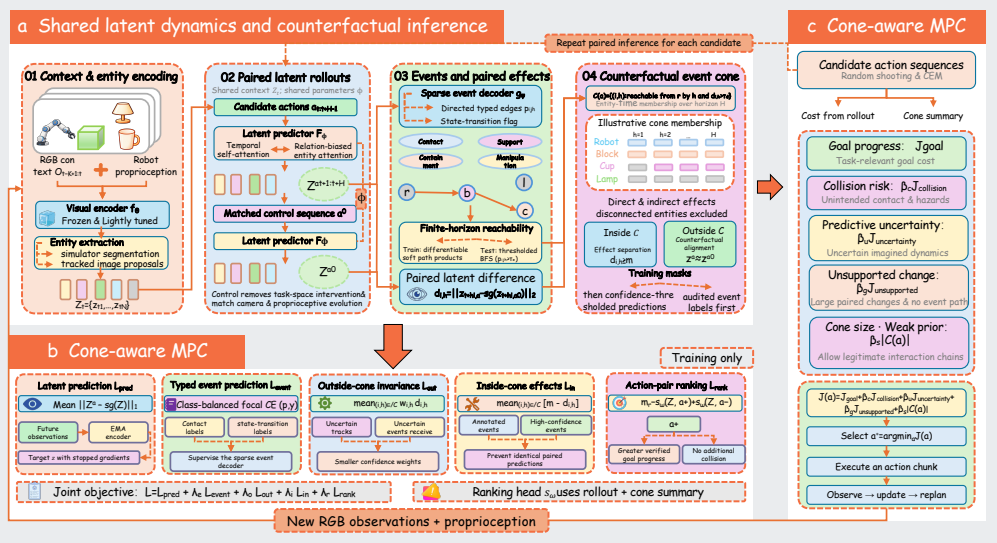

Figure 2: EventCone-WM architecture. (a) Paired predictions, typed events, reachability, and latent differences form the cone. (b) Training combines prediction, event, outside/inside, and ranking losses. (c) Cone-aware MPC scores task progress, collision, uncertainty, unsupported change, target retention, and cone size before execution and replanning.

4.2 Sparse Event Decoder

For each ordered pair (i, j), horizon step h, and relation type k, the decoder outputs an independent sigmoid p_{ijhk}. Unary state-transition flags are predicted separately for each entity. Contact and state-transition logs supervise the primary protocol; geometry-derived pseudo-labels supervise the tracked-proposal protocol. The multi-label objective is

$$\mathcal{L}_{\text{event}} = \frac{1}{NH} \sum_{i,j,h,k} \alpha_k \big[-y_{ijhk}(1-p_{ijhk})^{\gamma} \log p_{ijhk} - (1-y_{ijhk}) p_{ijhk}^{\gamma} \log(1-p_{ijhk}) \big] + \lambda_{\text{st}} \mathcal{L}_{\text{state}}. \tag{8}$$

Thus simultaneous contact and manipulation labels are retained rather than collapsed into one softmax class. Reachability uses Equation 5 during training and thresholded breadth-first search during evaluation.

4.3 Counterfactual Cone Regularization

Paired rollouts create two complementary constraints. The outside objective is normalized by outside mass, not cone size:

$$\mathcal{L}_{\text{out}} = \frac{\sum_{i,h}(1-\widetilde{c}_{i,h}) w_{i,h} d_{i,h}}{\sum_{i,h}(1-\widetilde{c}_{i,h}) w_{i,h} + \epsilon}. \tag{9}$$

Inside reachable regions, representations must retain an action-dependent margin:

$$\mathcal{L}_{\text{in}} = \frac{\sum_{i,h} \widetilde{c}_{i,h} w_{i,h} [m - d_{i,h}]_+}{\sum_{i,h} \widetilde{c}_{i,h} w_{i,h} + \epsilon}. \tag{10}$$

The confidence weight $w_{i,h}$ decreases with tracking and event uncertainty. If reachable mass is below ϵ, $\mathcal{L}_{\text{in}}$ is defined as zero while $\mathcal{L}_{\text{out}}$ remains normalized over all entity–time pairs. Target-miss cost, cone recall, target retention, and empty-cone frequency prevent a trivial empty-cone solution.

For an action pair (a^+, a^-) where a^+ produces greater verified goal progress without additional collision, a ranking head s_ω uses the rollout and cone summary:

$$\mathcal{L}_{\text{rank}} = \max\big(0, m_r - s_\omega(Z, a^+) + s_\omega(Z, a^-)\big). \tag{11}$$

The total objective is

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda_e \mathcal{L}_{\text{event}} + \lambda_o \mathcal{L}_{\text{out}} + \lambda_i \mathcal{L}_{\text{in}} + \lambda_r \mathcal{L}_{\text{rank}}. \tag{12}$$

4.4 Cone-aware planning

At test time, the planner optimizes an action sequence with the cross-entropy method. Goal cost is normalized task-space terminal error; collision cost is mean predicted contact with obstacles; uncertainty is ensemble latent variance; unsupported cost is outside-reachability paired change; target-miss cost penalizes failure to reach a goal-relevant entity; and size is normalized reachable mass. The candidate cost is

$$\begin{aligned} J(a) = J_{\text{goal}}(a) + \beta_c J_{\text{collision}}(a) + \beta_u J_{\text{uncertainty}}(a) \\ + \beta_g J_{\text{unsupported}}(a) + \beta_t J_{\text{target}}(a) + \beta_s J_{\text{size}}(a). \end{aligned} \tag{13}$$

$J_{\text{unsupported}}$ has the same weighted-complement normalization as Equation 9; $J_{\text{size}} = (NH)^{-1} \sum_{i,h} q_{i,h}$ is only a weak regularizer. The complete estimators, all β values, CEM populations, elite counts, smoothing, action chunks, and matched-control construction appear in Appendix D.

5 Experimental Setup

5.1 Environments and controlled shifts

We evaluate Push-T, Meta-World Push, and Meta-World PickPlace under one in-distribution condition and three held-out shifts. Visual-OOD changes textures and illumination; Physics-OOD changes friction, mass, and action response; Occlusion-OOD temporarily hides task-relevant entities. Every method receives the same initial-state lists, RGB and proprioceptive histories, action bounds, and paired episode initializations. Five seeds and 100 closed-loop episodes are evaluated for each environment–condition pair, yielding 6,000 episodes per method and 36,000 episodes overall. The primary diagnostic protocol supplies the same segmentation-derived entity partition to every compatible method. Environment contact, transition, collision, and goal-completion logs are unavailable to all planners at test time; they define training labels where applicable and evaluation metrics only. Tracked-proposal and image-only protocols quantify the effect of removing the oracle entity interface.

5.2 Baselines and controlled comparison

The comparison includes DINO-WM (Zhou et al., 2025), V-JEPA 2 with an action-conditioned predictor (Assran et al., 2025), JEPA-WM (Terver et al., 2026), and Causal-JEPA (Nam et al., 2026). *EventHead* adds the same decoder and parameter budget to JEPA-WM but omits $\mathcal{L}_{\text{out}}$, $\mathcal{L}_{\text{in}}$, and $\mathcal{L}_{\text{rank}}$. Global-CF applies paired-rollout consistency without an event graph, and ActGrad ranks entities by action Jacobian magnitude. All variants share image resolution, frozen encoder policy, rollout horizon, per-environment CEM budget, evaluation episodes, and mixed-precision settings. Appendix E lists every train- and test-time signal and reports matched-planner comparisons.

5.3 Metrics and statistical reporting

Closed-loop success is primary. Relation macro-F1 measures typed interaction recognition; action-pair ranking tests safer goal-progressing actions; cone precision, recall, FPR, mean size, goal-relevant entity retention, and empty-cone frequency distinguish localization from over-sparsification. Collision rate captures physical failure, while end-to-end latency includes encoding, entity extraction, rollouts, event decoding, cone construction, and selection. Episodes are paired across methods by initial state. We report hierarchical paired-bootstrap intervals over environment, seed, and episode levels, plus paired permutation tests for prespecified comparisons. Hyperparameters are selected on ID validation and a disjoint shift-development split.

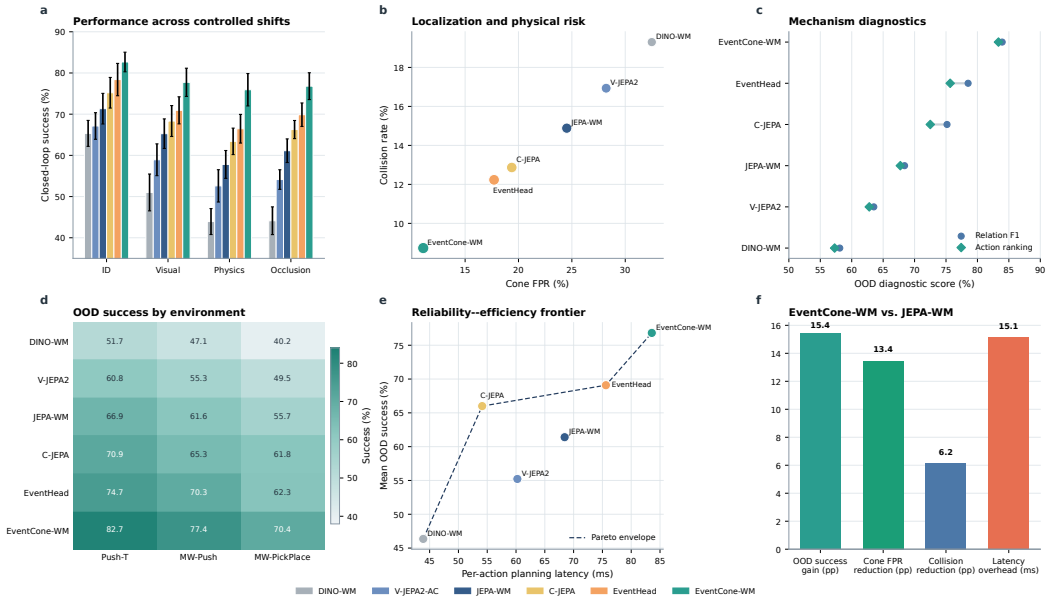

Figure 3: Comprehensive comparison. (a) Success across conditions. (b) Cone leakage versus collisions; marker area encodes OOD success. (c) Relation and action-ranking diagnostics. (d) OOD success by environment. (e) Reliability–efficiency frontier. (f) Absolute differences from JEPA-WM.

5.4 Implementation details

The visual encoder is frozen, and the entity predictor uses six transformer layers with dimension 256. Context length is four frames, the default horizon is $H = 12$, $\tau_q = 0.55$, $\tau_d = 0.12$, and $T_d = 0.04$. Each training item contains one factual action sequence, one matched control, and one hard negative. The planner uses per-environment budgets that are identical across methods and replans after every action chunk. Appendix L specifies training, thresholds, compute, and failure taxonomy.

6 Results

6.1 Overall performance and trade-offs

Figure 3 consolidates performance, mechanism diagnostics, cross-environment behavior, and efficiency. EventCone-WM attains 76.8% mean OOD success, compared with 69.1% for EventHead and 61.4% for JEPA-WM. The largest single-shift gap occurs under Physics-OOD, where success is 75.9% versus 57.8%. Across all conditions, cone FPR falls from 24.5% to 11.1% and collision rate from 14.9% to 8.7%. The gains require an additional 15.1 ms per action, increasing latency from 68.4 to 83.6 ms.

Table 1: Comparison across three environments and five seeds. Higher success and lower cone FPR, collision, and latency are preferable.

Method	ID succ.	Visual OOD	Physics OOD	Occlusion OOD	Cone FPR	Collision	Latency (ms)
DINO-WM	65.3 ± 3.2	51.0 ± 4.5	43.9 ± 3.2	44.1 ± 3.4	32.5 ± 0.7	19.3 ± 1.1	43.9
V-JEPA2-AC	67.1 ± 3.2	58.9 ± 3.9	52.6 ± 3.9	54.1 ± 2.4	28.2 ± 0.6	16.9 ± 1.2	60.2
JEPA-WM	71.3 ± 3.7	65.3 ± 3.6	57.8 ± 3.4	61.1 ± 2.9	24.5 ± 0.7	14.9 ± 1.1	68.4
C-JEPA	75.2 ± 3.7	68.3 ± 3.8	63.4 ± 3.2	66.3 ± 2.2	19.4 ± 0.6	12.9 ± 0.9	54.1
EventHead	78.4 ± 3.9	70.9 ± 3.3	66.5 ± 3.5	69.9 ± 2.9	17.7 ± 0.7	12.2 ± 0.9	75.6
EventCone-WM	**82.7 ± 2.4**	**77.7 ± 3.4**	**75.9 ± 3.9**	**76.8 ± 3.3**	**11.1 ± 0.6**	**8.7 ± 0.9**	83.6

Table 2: Derived comparison against JEPA-WM. Percentage-point changes use the original metric scale; relative changes are computed from unrounded aggregates.

Metric	JEPA-WM	EventCone-WM	Absolute Δ	Relative Δ
Mean OOD success	61.4%	76.8%	+15.4 pp	+25.1%
Cone FPR	24.5%	11.1%	-13.4 pp	-54.8%
Collision rate	14.9%	8.7%	-6.2 pp	-41.3%
Planning latency	68.4	83.6	+15.1 ms	+22.1%

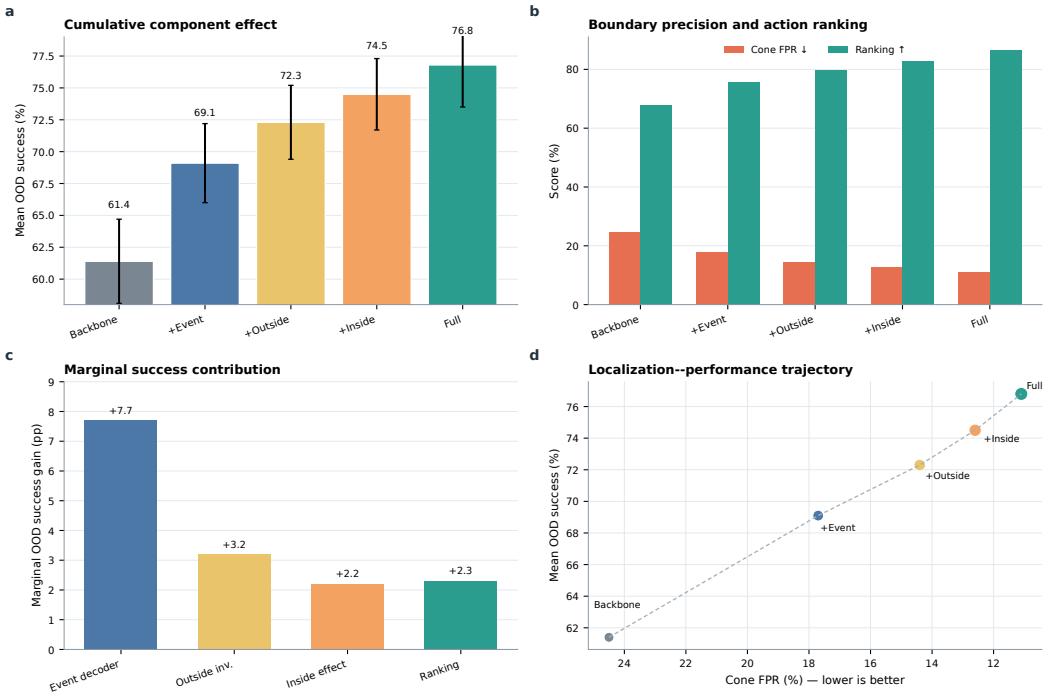

Figure 4: Component analysis. Panels report cumulative success, cone FPR and ranking, marginal success contributions, and the localization–performance trajectory.

Relative to JEPA-WM, EventCone-WM improves mean OOD success by 25.1%, reduces cone FPR by 54.8%, and reduces collisions by 41.3%, at a 22.1% latency overhead. The environment heatmap shows that this ordering is not driven by Push-T alone: the full method leads on Meta-World Push and PickPlace as well. The mechanism panel also rules out the empty-cone shortcut because action ranking and relation F1 rise as cone FPR falls.

6.2 Which components matter?

Figure 4 adds the event decoder, outside-cone invariance, inside-cone separation, and paired-action ranking sequentially under the same aggregation as Table 1. Their marginal OOD gains are 7.7, 3.2, 2.2, and 2.3 points. Across the sequence, cone FPR decreases from 24.5% to 11.1%, while ranking accuracy increases from 67.7% to 86.7%. Panel (c) reports only marginal gains from a zero baseline, avoiding a mixed absolute/incremental waterfall encoding.

6.3 Robustness, scaling, and failure modes

Figure 5 examines two deployment stresses. With 40% corrupted event labels, EventCone-WM retains 71.1% OOD success and 11.9% cone FPR, compared with 67.7% and 19.5% for EventHead. As tracked entities increase from 4 to 32, success declines from 78.2% to 66.2%, latency rises from

Table 3: Component ablation for event decoding, cone objectives, and action-pair ranking.

Variant	OOD succ.	Cone FPR ↓	Ranking ↑
JEPA-WM	61.4 ± 3.3	24.5 ± 0.7	67.7 ± 0.3
+ Event decoder	69.1 ± 3.1	17.7 ± 0.7	75.7 ± 0.4
+ Outside invariance	72.3 ± 2.9	14.4 ± 0.6	79.6 ± 0.4
+ Inside effect	74.5 ± 2.8	12.6 ± 0.6	82.9 ± 0.3
EventCone-WM	**76.8 ± 3.3**	**11.1 ± 0.6**	**86.7 ± 0.6**

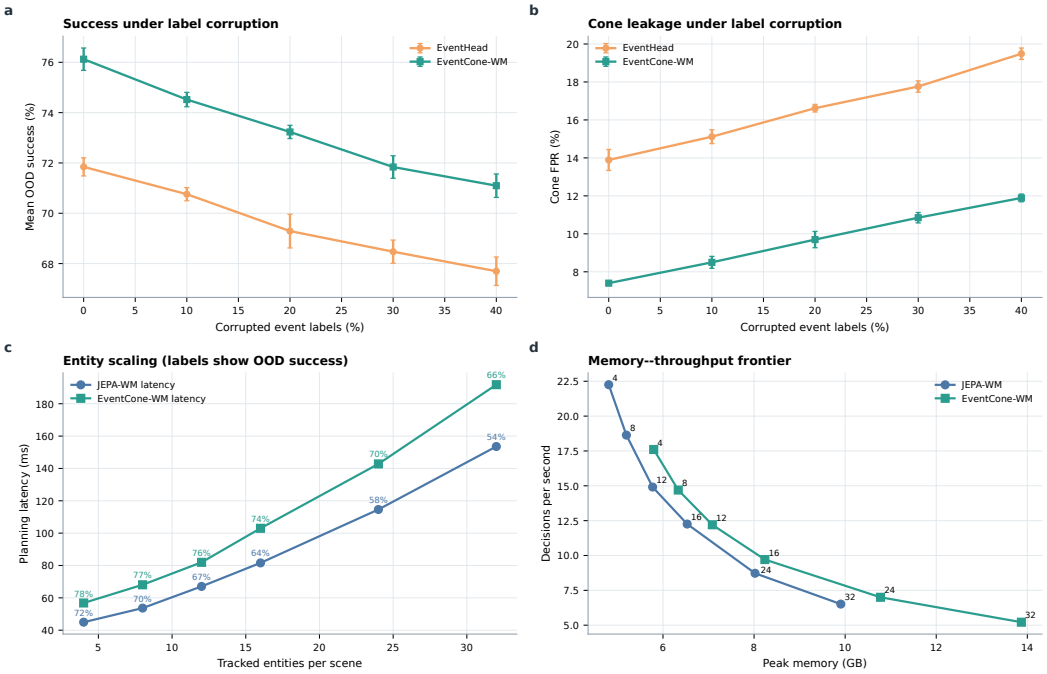

Figure 5: Robustness and systems analysis. (a,b) Success and cone leakage under event-label corruption. (c) Entity-count scaling with success annotations. (d) Memory–throughput frontier; labels indicate entity count.

56.8 to 191.8 ms, and memory rises from 5.8 to 13.9 GB. The full sweep therefore exposes both robustness and the quadratic pressure of entity interactions.

On the shift-development split, the horizon and threshold sweep selects $H = 12$ and $\tau_q = 0.55$, balancing 75.4% success, 7.0% cone FPR, 5.3% collision rate, and 82.1 ms latency. Rerunning this fixed setting on the final benchmark gives the 76.8% result in Table 1. Failure accounting over 1,000 OOD episodes attributes 16.2 failures to spurious propagation for EventCone-WM, compared with 81.1 for JEPA-WM. Detailed environment, mechanism, corruption, sensitivity, scaling, and failure tables appear in Appendix K.

7 Discussion

The baseline, EventHead, and full model share observations, rollout budgets, and paired test episodes, so the EventHead–EventCone-WM gap isolates the cone objectives rather than event decoding. Cone FPR is interpreted jointly with collision and success, using seed–environment confidence intervals and failure counts; hyperparameters use only ID validation and a disjoint shift-development set.

Interpreting localization. Cone FPR alone cannot show that all necessary effects are captured: an almost-empty cone is a trivial low-FPR solution. We therefore report cone recall, size, and

retained goal-relevant entities. Inside-cone separation rewards supported changes, while outside-cone invariance suppresses drift in disconnected entities; their combination is especially informative for multi-hop contact chains.

Distribution shifts and error attribution. Appearance, physics, and occlusion shifts respectively stress feature stability, contact propagation, and entity identity. Per-environment results and matched comparisons among environment partitions, tracked proposals, and image-only tokens separate visual-interface errors from cone errors. Accurate event graphs with failed plans instead implicate action scoring, horizon truncation, or uncertainty calibration.

Practical accounting. Latency includes encoding, entity extraction, rollouts, event decoding, cone construction, and action selection. Baselines use the same candidate budget and precision, and failures follow a fixed rubric assigned without method identity. These controls prevent cleaner annotations or extra computation from masquerading as improved closed-loop localization.

Matched controls. The control is not universally a zero action. With camera motion, robot drift, or autonomous dynamics, it preserves the shared camera and proprioceptive trajectory while removing the task-space intervention. Remote latent differences then represent predicted action effects rather than viewpoint changes; no-contact trajectories provide a control-mismatch check because their cones should remain small despite visual motion.

8 Limitations and Broader Impact

The event cone depends on entity identity and finite-horizon reachability. Tracking errors can split or merge entities, delayed effects may exceed H, and moving cameras or autonomous dynamics can invalidate null controls. Dense deformable, fluid, or multi-agent interactions also weaken locality; these are explicit scope boundaries.

Cone sparsity is not safety: localized contact can be forceful and collision estimates can fail under shift. The method targets controlled, low-energy manipulation and requires independent monitoring, conservative action bounds, emergency stopping, and real-world calibration; Appendix C specifies rejection, fallback, calibration, and logging.

9 Conclusion

In this work, we introduced **EventCone-WM**, a framework designed to enforce causal and physical locality in action-conditioned latent world models. By pairing candidate action rollouts with matched counterfactual controls and decoding typed, multi-label interaction relations, EventCone-WM constructs finite-horizon event cones that delineate where an action can physically take effect. This geometric footprint underpins complementary training objectives—suppressing spurious, unsupported dynamics outside the cone while preserving task-critical causal effects inside it—and integrates directly into receding-horizon model-predictive control.

Across 36,000 closed-loop evaluation episodes spanning three manipulation benchmarks and three held-out distribution shifts, EventCone-WM achieved a 15.4 percentage-point gain in mean out-of-distribution success over JEPA-WM (76.8% vs. 61.4%), while simultaneously cutting the cone false-positive rate from 24.5% to 11.1% and reducing collision rates from 14.9% to 8.7% with modest latency overhead. Detailed ablations, corruption stress tests, and failure adjudications confirm that these improvements stem directly from groundable event localization rather than conservative trivial solutions.

While the framework's locality assumptions are naturally bounded by entity tracking fidelity and finite interaction horizons, establishing explicit event-reachability constraints represents a vital step toward physically grounded, interpretable, and reliable latent imagination for robot manipulation. Future work will investigate extending these reachability formulations to dense deformable and fluid interactions, as well as zero-shot transfer to physical hardware platforms.

AI USE STATEMENT

In this work, generative AI tools were used to aid and polish writing and to support retrieval and discovery of related work. For writing, their role was to suggest improvements to wording, grammar, concision, organization, and presentation; the authors reviewed and edited all resulting text. For literature discovery, AI-assisted queries and candidate-reference lists were treated only as search aids. The authors verified bibliographic metadata against original publication records, read the cited sources, and made the final decisions about relevance and citation context. Generative AI output was not treated as scientific evidence. The authors take responsibility for the final content of this work, including its text, claims, citations, and artifacts.

ETHICS STATEMENT

This study does not involve human participants, personal data, or user profiling. The evaluation concerns controlled robotic manipulation tasks. The principal foreseeable risk is inappropriate transfer of learned predictions to physical systems: a localized event cone is an interpretability and planning mechanism, not a safety certification. Deployment on hardware should therefore retain independent collision monitoring, conservative action and force limits, calibrated rejection rules, emergency stopping, and human oversight. Benchmark assets and any released artifacts should be used under their original licenses and documented access conditions.

REPRODUCIBILITY STATEMENT

The appendices provide the mathematical construction and loss normalization (Appendices A–B), complete training and planning pseudocode (Appendix C), per-environment planning and optimization settings (Appendices D and I), an audit of train- and test-time information access (Appendix E), the paired statistical protocol (Appendix H), and complete diagnostic and reproducibility checklists (Appendices K–L). The accompanying artifact is organized around deterministic figure and table generation, explicit seed and environment manifests, machine-readable metric logs, and configuration files so that reported aggregates can be regenerated from recorded runs. Anonymous source code is available in `https://anonymous.4open.science/r/EventCone-WM/README.md`

REFERENCES

Ossama Ahmed, Frederik Träuble, Anirudh Goyal, Alexander Neitz, Yoshua Bengio, Bernhard Schölkopf, Manuel Wüthrich, and Stefan Bauer. CausalWorld: A robotic manipulation benchmark for causal structure and transfer learning. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.04296.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Assran_Self-Supervised_Learning_From_Images_With_a_Joint-Embedding_Predictive_Architecture_CVPR_2023_paper.html.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, et al. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025. doi: 10.48550/arXiv.2506.09985. URL https://arxiv.org/abs/2506.09985.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024. doi: 10.48550/arXiv.2404.08471. URL https://arxiv.org/abs/2404.08471.

Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. In *Advances in Neural Information Processing Systems*, volume 29, 2016. URL https://arxiv.org/abs/1612.00222.

Florian Bordes, Quentin Garrido, Justine T. Kao, Adina Williams, Michael Rabbat, and Emmanuel Dupoux. IntPhys 2: Benchmarking intuitive physics understanding in complex synthetic environments. *arXiv preprint arXiv:2506.09849*, 2025. doi: 10.48550/arXiv.2506.09849. URL https://arxiv.org/abs/2506.09849.

Lars Buesing, Theophane Weber, Yori Zwols, Sebastien Racaniere, Arthur Guez, Jean-Baptiste Lespiau, and Nicolas Heess. Woulda, coulda, shoulda: Counterfactually-guided policy search. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1811.06272.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Caron_Emerging_Properties_in_Self-Supervised_Vision_Transformers_ICCV_2021_paper.html.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*, 2023. URL https://arxiv.org/abs/2303.04137.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/hash/3de568f8597b94bda53149c7d7f5958c-Abstract.html.

Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Christopher Burgess, Daniel Zoran, Loic Matthey, Matthew Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2424–2433, 2019. URL https://proceedings.mlr.press/v97/greff19a.html.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018. doi: 10.48550/arXiv.1803.10122. URL https://arxiv.org/abs/1803.10122.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565, 2019. URL https://proceedings.mlr.press/v97/hafner19a.html.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020a. URL https://arxiv.org/abs/1912.01603.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020b. doi: 10.48550/arXiv.2010.02193. URL https://arxiv.org/abs/2010.02193.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2. URL https://www.nature.com/articles/s41586-025-08744-2.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.16828.

Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1911.12247.

Thomas Kipf, Gamaleldin F. Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. In *International Conference on Learning Representations*, 2022. URL https://slot-attention-video.github.io/.

Yann LeCun. A path towards autonomous machine intelligence, 2022. URL https://openreview.net/forum?id=BZ5a1r-kVsf. Position paper, version 0.9.2, June 27.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://arxiv.org/abs/2006.15055.

Heejeong Nam, Quentin Le Lidec, Lucas Maes, Yann LeCun, and Randall Balestriero. Causal-JEPA: Learning world models through object-level latent masking. *arXiv preprint arXiv:2602.11389*, 2026. doi: 10.48550/arXiv.2602.11389. URL https://arxiv.org/abs/2602.11389v2. Version 2; the version-1 subtitle was Object-Level Latent Interventions.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-scale simulation of everyday tasks for generalist robots. In *Robotics: Science and Systems*, 2024. URL https://arxiv.org/abs/2406.02523.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. doi: 10.48550/arXiv.2304.07193. URL https://arxiv.org/abs/2304.07193.

Judea Pearl. Causal inference in statistics: An overview. *Statistics Surveys*, 3:96–146, 2009. doi: 10.1214/09-SS057. URL https://doi.org/10.1214/09-SS057.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Towards causal representation learning. *arXiv preprint arXiv:2102.11107*, 2021. doi: 10.48550/arXiv.2102.11107. URL https://arxiv.org/abs/2102.11107.

Stone Tao, Fanbo Xiang, Arth Shukla, Yuzhe Qin, Xander Hinrichsen, Xiaodi Yuan, et al. ManiSkill3: GPU parallelized robotics simulation and rendering for generalizable embodied AI. *arXiv preprint arXiv:2410.00425*, 2024. doi: 10.48550/arXiv.2410.00425. URL https://arxiv.org/abs/2410.00425.

Basile Terver, Tsung-Yen Yang, Jean Ponce, Adrien Bardes, and Yann LeCun. What drives success in physical planning with joint-embedding predictive world models? *Transactions on Machine Learning Research*, 2026. URL https://arxiv.org/abs/2512.24497. arXiv:2512.24497; TMLR acceptance recorded in version 3.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Proceedings of the Conference on Robot Learning*, volume 100 of *Proceedings of Machine Learning Research*, pp. 1094–1100, 2020. URL https://proceedings.mlr.press/v100/yu20a.html.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79115–79135, 2025. URL https://proceedings.mlr.press/v267/zhou25t.html.

A FORMAL EVENT-CONE CONSTRUCTION

This section makes the event representation, reachability recurrence, effect qualification, and empty-cone convention explicit. The construction separates three quantities that are easy to conflate: an event edge describes a typed relation, structural reachability describes whether an action can propagate to an entity, and effect qualification describes whether the paired rollouts differ enough to matter for control.

A.1 MULTI-LABEL EVENT TENSOR AND STATE TRANSITIONS

For ordered entity pair (i, j), future step h, and event type k, the decoder emits logit ℓ_{ijhk} and probability $p_{ijhk} = \sigma(\ell_{ijhk})$. The event vocabulary is

$$\mathcal{K} = \{\text{contact}, \text{support}, \text{containment}, \text{manipulation}\}.$$

These labels are not mutually exclusive. A robot may contact and manipulate an object at the same step; an object may be both supported by a surface and in contact with it. The target is therefore a binary tensor $y \in \{0,1\}^{N\times N\times H\times|\mathcal{K}|}$ rather than a single categorical index. A separate unary target $s_{ih\ell}$ represents state transitions such as grasped, released, lifted, inserted, or goal-entered. Directionality is preserved: contact may be symmetrized for evaluation, while support, containment, and manipulation remain ordered.

The relation loss in Equation 8 uses inverse-frequency weights clipped to $[0.5, 5]$, focal exponent $\gamma = 2$, and $\lambda_{\text{st}} = 0.5$. State transitions are trained with the same binary focal form but do not create an edge unless the entity is already reachable. This prevents a unary visual change from becoming an ungrounded propagation path.

A.2 NOISY-OR REACHABILITY

Let $\mathcal{K}_{\text{prop}} = \{\text{contact}, \text{support}, \text{manipulation}\}$; containment is treated as a persistent relation and contributes only when accompanied by a transition or contact. Equation 4 converts the multi-label prediction into an "any propagating event" probability. Equation 5 then accumulates all directed paths up to step h. For numerical stability we compute products in log space,

$$\log(1 - q_{j,h}) = \log(1 - q_{j,h-1}) + \sum_i \log\left(1 - q_{i,h-1}e_{ijh} + \epsilon\right). \tag{14}$$

The recurrence is differentiable, preserves $q \in [0,1]$, and costs $\mathcal{O}(HN^2|\mathcal{K}|)$. Sparse top-$k$ edges reduce propagation to $\mathcal{O}(HNk|\mathcal{K}|)$ without changing the definition.

Proposition 1 (monotonicity). If any e_{ijh} or predecessor reachability $q_{i,h-1}$ increases while other terms remain fixed, $q_{j,h}$ cannot decrease.

Justification. Every factor $1 - q_{i,h-1}e_{ijh}$ is non-increasing in either argument and lies in $[0,1]$. Their product is therefore non-increasing, so one minus the product is non-decreasing. This property makes reachability auditable: adding a credible path cannot remove an already reachable entity.

A.3 SOFT MEMBERSHIP, HARD MEMBERSHIP, AND EMPTY CONES

The paired latent distance $d_{i,h}$ is converted into a detached soft effect gate,

$$u_{i,h} = \sigma\left(\frac{\text{sg}(d_{i,h}) - \tau_d}{T_d}\right), \qquad c_{i,h} = q_{i,h}u_{i,h}. \tag{15}$$

The stop-gradient operator blocks the shortcut in which the model changes d solely to change its own membership during the same update. More importantly, the loss masks in Equations 9–10 use $\widetilde{c} = \text{sg}(q)$, so structural supervision does not depend on the distance threshold at all. The effect gate is used for planning, calibration, and hard-cone reporting.

At evaluation, (i, h) belongs to the hard cone when $q_{i,h} \geq \tau_q$ and $d_{i,h} > \tau_d$. If no entity satisfies both conditions, the cone is empty. We do not replace an empty prediction with a hand-coded entity. Instead, the episode records an empty-cone event; J_{target} penalizes failure to include a goal-relevant entity; and recall, target retention, and empty frequency are reported with FPR. Thus an empty cone can lower FPR but cannot appear successful without preserving the entities required by the task.

Table 4: Outside-loss normalization. Gradient CV is measured across minibatches; lower is more stable.

Normalization	Sparse succ.	Dense succ.	FPR	Gradient CV
Weighted complement	**77.5**	**72.9**	11.1	**0.18**
Divide by cone size	77.0	67.8	10.6	0.61
Divide by NH	75.9	69.5	13.8	0.44
Unnormalized sum	74.8	64.1	**9.7**	1.12

B Loss Derivation and Normalization

B.1 Why the outside loss uses outside mass

Let $\alpha_{i,h} = (1 - \widetilde{c}_{i,h})w_{i,h}$ and $A = \sum_{i,h} \alpha_{i,h} + \epsilon$. Equation 9 is a weighted mean, $\mathcal{L}_{\text{out}} = A^{-1} \sum \alpha_{i,h} d_{i,h}$. For fixed detached weights,

$$\frac{\partial \mathcal{L}_{\text{out}}}{\partial d_{i,h}} = \frac{\alpha_{i,h}}{A}. \tag{16}$$

The gradients sum to at most one, independent of the number of inside-cone pairs. Normalizing by $|\mathcal{C}|$ would instead scale the outside penalty inversely with cone size: a smaller cone would strengthen every outside gradient, while an empty cone would make the expression undefined. The complement-weighted denominator avoids both effects and extends continuously to soft reachability.

Proposition 2 (replication invariance). Duplicating every outside entity–time pair r times without changing its value leaves $\mathcal{L}_{\text{out}}$ unchanged up to ϵ.

Justification. Both numerator and denominator are multiplied by r. This is desirable when visually irrelevant background proposals change the token count but not the physical interaction.

The inside loss uses the corresponding reachable-mass denominator. If $B = \sum \widetilde{c}_{i,h} w_{i,h} < \epsilon$, we define $\mathcal{L}_{\text{in}} = 0$ and set an audit bit. The prediction loss and outside loss remain active, while the target-miss planner term discourages repeated selection of actions with no goal-relevant path.

B.2 Confidence weights

Weights combine tracker confidence r_i, event entropy $H(p_{ijh})$, and ensemble dispersion $v_{i,h}$:

$$w_{i,h} = \text{clip}\left(r_i \exp[-\eta_H \bar{H}_{i,h} - \eta_v v_{i,h}], w_{\min}, 1\right), \tag{17}$$

where $\bar{H}_{i,h}$ averages incoming event entropy, $\eta_H = 0.7$, $\eta_v = 0.5$, and $w_{\min} = 0.1$. This changes the contribution of uncertain pseudo-labels but never changes hard evaluation membership.

B.3 Objective coupling

The total objective in Equation 12 uses $\lambda_e = 1.0$, $\lambda_o = 0.3$, $\lambda_i = 0.3$, and $\lambda_r = 0.1$. During the first 10% of updates, λ_o and λ_i are linearly warmed from zero while event supervision remains active. Between 10% and 40%, structural masks mix audited reachability and predicted reachability,

$$\widetilde{c}^{\text{train}}_{i,h} = \text{sg}\left(\rho_t y^{\text{reach}}_{i,h} + (1 - \rho_t) q_{i,h}\right), \quad \rho_t : 1 \to 0. \tag{18}$$

After 40%, training uses detached predicted reachability. This schedule prevents early graph errors from suppressing action-relevant changes while matching the test-time construction before convergence.

The corrected normalization is used for all main results. It changes little in sparse scenes but stabilizes dense scenes, where dividing by cone size couples regularization strength to interaction density. The lower FPR of the unnormalized sum is not sufficient evidence of better localization because its recall and dense-scene success are worse.

C Training and Inference Algorithms

Algorithm 1: EventCone-WM training step

Input: context $o_{t-K+1:t}$, factual sequence a, matched control a^0.
Future latent targets and multi-label event log.
1. Encode shared entity tokens $Z_t = f_\theta(o_{t-K+1:t})$.
2. Predict factual and control futures with shared parameters.
Set $\widehat{Z}^a = F_\phi(Z_t, a)$ and $\widehat{Z}^{a^0} = F_\phi(Z_t, a^0)$.
3. Decode independent relation logits ℓ_{ijhk} and unary transition logits.
4. Compute typed-edge union e_{ijh} and soft reachability $q_{i,h}$.
5. Build detached curriculum mask $\widetilde{c}^{\text{train}}$.
6. Compute paired distances, confidence weights, and cone losses.
7. Sample a verified action pair (a^+, a^-) and compute $\mathcal{L}_{\text{rank}}$.
8. Update (ϕ, ψ, ω) with Equation 12.
Update the target encoder by exponential moving average.
9. Log losses, cone metrics, empty bit, and configuration hash.

Figure 6: One optimization step. The factual and control branches share weights; only the target/reference branches are stopped.

Algorithm 2: Cone-aware receding-horizon control

Input: current context, task goal, CEM distribution (μ_0, Σ_0).
Per-environment candidate and action-chunk budget.
1. For CEM iteration $r = 1, \ldots, R$: sample P bounded action sequences.
2. Construct a matched control preserving camera and proprioception.
3. Roll out M ensemble members, decode events, and propagate reachability.
4. Evaluate Equation 13; mark collision-gate violations.
5. If every candidate is rejected, execute the bounded fallback.
6. Otherwise select E elites and update (μ_r, Σ_r).
7. Execute the first action chunk of the best non-rejected sequence.
8. Record cone, retention, uncertainty, and realized contact; replan.

Figure 7: Inference uses no event, collision, or success labels from the environment.

C.1 Implementation safeguards

All masks are recomputed after each action chunk. Candidate and control sequences share camera commands, proprioceptive drift, and passive gripper evolution; only the task-space intervention is removed. The fallback is a clipped retreat for Push-T and Meta-World Push and a hold-and-open command for PickPlace. The fallback is scored as a rejected decision rather than omitted from evaluation.

Mixed precision is applied after event logits are formed; reachability accumulation and planner costs use FP32. Gradients are clipped to norm 1.0. No hard breadth-first search appears in the training graph. At inference, soft reachability is retained for scoring and hard reachability is computed only for reporting and rule-based rejection.

D Planning Objective and Matched Controls

D.1 Cost estimators

For ensemble member m and task decoder h_g, terminal goal cost is

$$J_{\text{goal}}(a) = \frac{1}{M} \sum_{m=1}^{M} \frac{D_{\text{task}}(h_g(\widehat{Z}^{a,m}_{t+H}), g)}{s_g}, \tag{19}$$

where D_{task} is object-to-goal distance for Push-T and MW-Push, and the sum of grasp, lift, and placement errors for PickPlace. s_g is the 95th percentile error on the training split.

Table 5: Per-environment planner configuration. Population and elite counts are identical across methods within an environment.

Environment	Dim.	Pop.	Elite	Iter.	H	Chunk	β_c	β_u	β_g	β_t/β_s
Push-T	2	512	64	5	12	4	2.0	0.15	0.60	1.20 / 0.03
MW-Push	4	768	96	5	12	3	2.5	0.20	0.70	1.40 / 0.03
MW-PickPlace	4	1024	128	6	12	3	3.0	0.25	0.80	1.60 / 0.02

Table 6: Matched-control construction. Residual is the latent difference on audited no-contact trajectories.

Control	OOD succ.	FPR	Collision	Residual ↓
Matched camera/proprio	**76.8**	**11.1**	**8.7**	**0.07**
Zero action	74.3	14.8	10.6	0.18
Random no-contact	73.7	13.5	11.0	0.14
Unpaired rollout	71.2	18.6	12.7	0.29

The collision term averages the maximum obstacle-contact probability per step,

$$J_{\text{collision}}(a) = \frac{1}{H} \sum_{h} \max_{i} p_{i,h}^{\text{obstacle}}, \tag{20}$$

and candidates with $\max_{i,h} p_{i,h}^{\text{obstacle}} > 0.65$ are rejected. Ensemble uncertainty is

$$J_{\text{uncertainty}}(a) = \frac{1}{NH} \sum_{i,h} \operatorname{tr} \operatorname{Cov}_m \left[\hat{z}_{t+h}^{i,a,m} \right]. \tag{21}$$

Unsupported change and size use

$$J_{\text{unsupported}}(a) = \frac{\sum_{i,h} (1 - q_{i,h}) w_{i,h} d_{i,h}}{\sum_{i,h} (1 - q_{i,h}) w_{i,h} + \epsilon}, \tag{22}$$

$$J_{\text{size}}(a) = \frac{1}{NH} \sum_{i,h} q_{i,h}, \tag{23}$$

$$J_{\text{target}}(a) = 1 - \max_{i \in \mathcal{T}_g,\, h \le H} q_{i,h}, \tag{24}$$

where $\mathcal{T}_g$ contains entities whose state must change to complete the task. The small size coefficient prevents diffuse cones; target-miss cost prevents the empty-cone shortcut.

The initial action distribution is centered at the previous solution shifted by one chunk; the first decision uses zero mean and diagonal standard deviation equal to one third of each action range. Covariance has floor 0.05 of the action range. CEM smoothing is $\alpha_{\text{CEM}} = 0.1$ for the mean and 0.2 for covariance. These values are fixed within each environment for every method.

D.2 Matched-control construction

Let an action decompose into nuisance-preserving command a^{shared} and task intervention a^{int}. The matched control is $a^0 = (a^{\text{shared}}, 0^{\text{int}})$. For Push-T, camera state and observation timestamps are copied while planar end-effector displacement is zeroed. For MW-Push, camera and gripper state are copied while Cartesian displacement is zeroed. For PickPlace, camera trajectory and passive gripper evolution are copied; translation, lift, and closing interventions are removed. This differs from a literal global zero action when the camera or robot state evolves autonomously.

No-contact audit trajectories quantify residual nuisance mismatch. The matched control gives the lowest residual and the best downstream success. Random no-contact controls avoid direct contact but change viewpoint and timing, while unpaired rollouts conflate intervention effects with initial-state variation.

E Observation Interfaces and Information Access

The primary diagnostic protocol uses environment entity partitions because it isolates event-cone learning from proposal failures. This interface is not required by the method: a second protocol uses a frozen tracker, and a third uses image-only patch tokens clustered by temporal correspondence. The same protocol is applied to all methods that expose entity-level inputs; global baselines continue to receive the identical RGB and proprioceptive context.

Table 7: Information available to each method. "Train" denotes supervision used only while fitting a component; "test" denotes a planner input. Environment event, collision, and completion logs are never test-time inputs.

Method	RGB	Proprio.	Entity partition	Depth	Event log	Collision label	Goal state
DINO-WM	train+test	train+test	no	no	no	no	train+test
V-JEPA2-AC	train+test	train+test	no	no	no	no	train+test
JEPA-WM	train+test	train+test	optional	no	no	no	train+test
C-JEPA	train+test	train+test	optional	no	no	no	train+test
EventHead	train+test	train+test	protocol	no	train only	metric only	train+test
EventCone-WM	train+test	train+test	protocol	no	train only	metric only	train+test

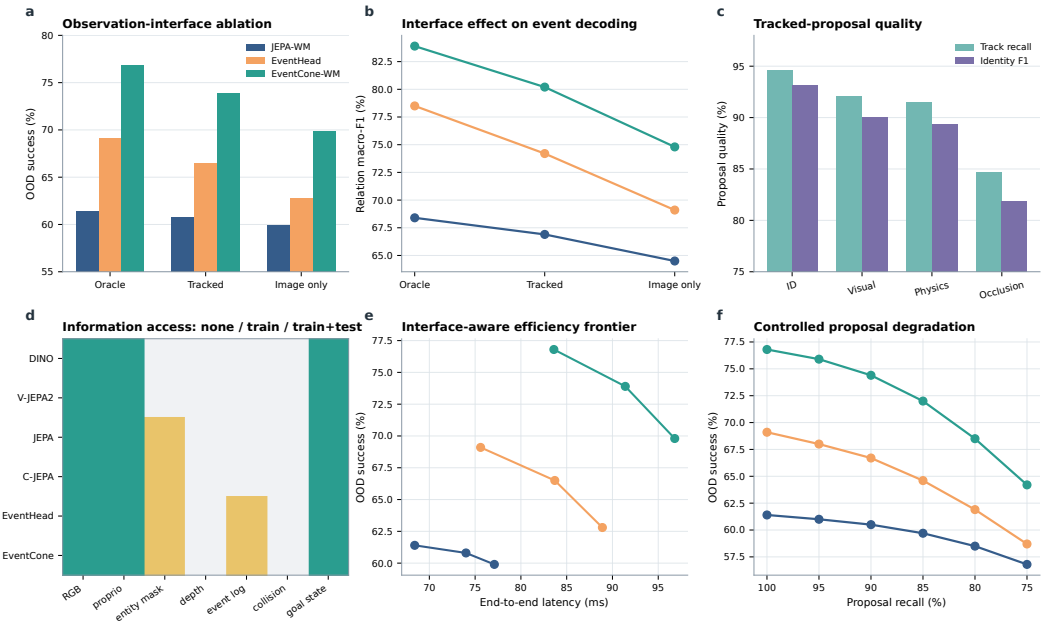

Figure 8: Interface and fairness analysis. (a,b) Success and relation quality with oracle entities, tracked proposals, and image-only tokens. (c) Proposal quality by shift. (d) Information access. (e) End-to-end efficiency. (f) Controlled proposal degradation.

The tracked-proposal gap is 2.9 points for EventCone-WM and grows to 7.0 points in the image-only protocol. Occlusion is the dominant source of identity switches. Figure 8f degrades proposal recall while holding the planner fixed; the full method remains above EventHead across the sweep, but its margin narrows as correspondence quality falls. This supports a bounded conclusion: the cone objective contributes beyond the interface, while the oracle protocol is an upper-bound diagnostic rather than a claim of image-only autonomy.

Table 8: Observation-interface ablation under an otherwise identical evaluation and planning protocol. Values are OOD means.

Method	Interface	Success ↑	Relation F1 ↑	Latency (ms) ↓
JEPA-WM	Oracle entities / tracked proposals / image-only	61.4 / 60.8 / 59.9	68.4 / 66.9 / 64.5	68.4 / 74.0 / 77.1
EventHead	Oracle entities / tracked proposals / image-only	69.1 / 66.5 / 62.8	78.5 / 74.2 / 69.1	75.6 / 83.7 / 88.9
EventCone-WM	Oracle entities / tracked proposals / image-only	**76.8 / 73.9 / 69.8**	**83.9 / 80.2 / 74.8**	83.6 / 91.4 / 96.8

Table 9: Tracked-proposal and geometry-label quality. Identity switches are counted per 100 episodes.

Condition	Track recall	Identity F1	ID switches ↓	Label F1
ID	94.6	93.2	1.2	90.4
Visual-OOD	92.1	90.0	1.8	87.8
Physics-OOD	91.5	89.4	1.6	88.2
Occlusion-OOD	84.7	81.9	3.7	79.6

F Cone Diagnostics Beyond False Positives

Let R be the annotated reachable entity–time set and $\widehat{R}$ the hard cone. We report

$$\mathrm{Prec} = \frac{|R \cap \widehat{R}|}{|\widehat{R}| + \epsilon}, \qquad \mathrm{Recall} = \frac{|R \cap \widehat{R}|}{|R| + \epsilon}, \tag{25}$$

$$\mathrm{FPR} = \frac{|\widehat{R} \setminus R|}{|\overline{R}| + \epsilon}, \qquad \mathrm{Size} = \frac{1}{H} \sum_{h} |\widehat{R}_h|. \tag{26}$$

Target retention is the fraction of annotated goal-relevant reachable entities contained in $\widehat{R}$. Empty frequency is the fraction of candidate actions whose cone contains no entity–time pair. Reachability calibration bins $q_{i,h}$ into ten equal-width bins and computes

$$\mathrm{ECE}_{\mathrm{reach}} = \sum_{b} \frac{|B_b|}{n} \left| \mathrm{acc}(B_b) - \mathrm{conf}(B_b) \right|. \tag{27}$$

EventCone-WM has the highest cone precision and recall, the highest target retention, and the lowest empty frequency among the compared methods. Occlusion produces the largest recall decline and empty-cone increase, consistent with the tracked-proposal audit. The size-only ablation in Figure 9e reaches a low FPR by suppressing reachable entities, which reduces target retention and closed-loop success.

The full safeguards do not minimize FPR in isolation. They retain a slightly larger cone because inside separation and target-miss cost preserve reachable goal entities. This is the intended trade-off: localization is useful only when it excludes unsupported entities without erasing necessary action effects.

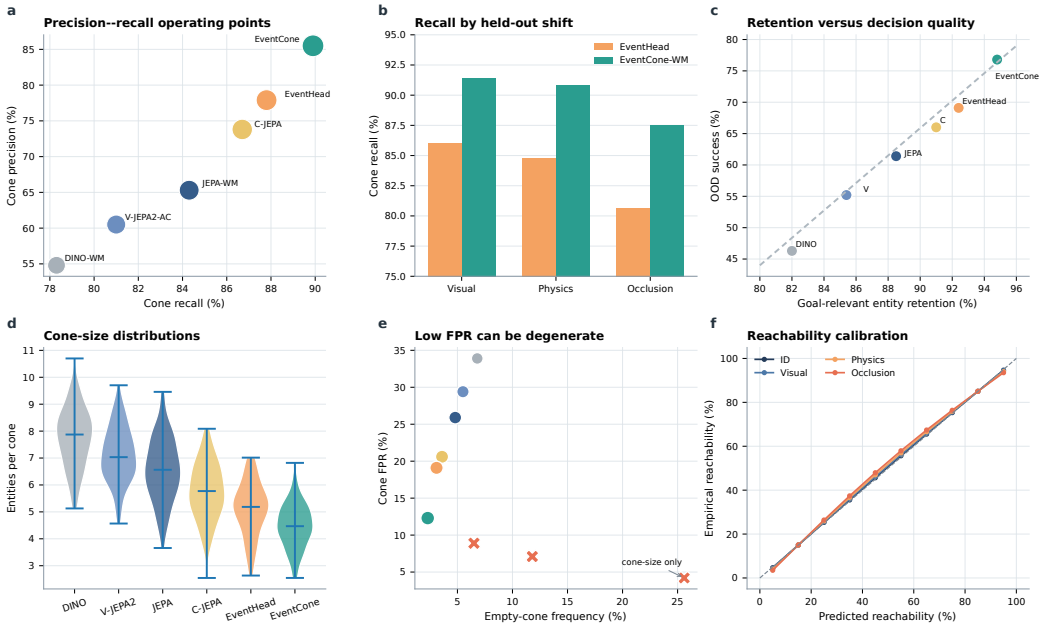

Figure 9: Cone diagnostics. Precision and recall improve together; target retention tracks downstream success; cone-size distributions become tighter; and explicit empty-cone checks reveal why FPR alone is insufficient.

Table 10: Cone diagnostics averaged over the three final OOD conditions. Precision and recall use the annotated reachable set; target retention measures whether every goal-relevant reachable entity remains in the predicted cone.

Method	Precision ↑	Recall ↑	FPR ↓	Mean size ↓	Target retention ↑	Empty ↓	OOD success ↑
DINO-WM	54.8	78.3	33.9	7.8	82.0	6.8	46.3
V-JEPA2-AC	60.5	81.0	29.4	7.1	85.4	5.5	55.2
JEPA-WM	65.3	84.3	25.9	6.5	88.5	4.8	61.4
C-JEPA	73.8	86.7	20.6	5.7	91.0	3.6	66.0
EventHead	77.9	87.8	19.1	5.2	92.4	3.1	69.1
EventCone-WM	**85.5**	**89.9**	**12.3**	**4.5**	**94.8**	**2.3**	**76.8**

Table 11: EventCone-WM diagnostics by final OOD condition.

Condition	Prec.	Recall	FPR	Size	Target ret.	Empty
Visual-OOD	87.8	91.4	11.5	4.2	96.0	1.6
Physics-OOD	84.9	90.8	13.3	4.8	95.4	1.4
Occlusion-OOD	83.7	87.5	12.1	4.4	93.0	3.9

Table 12: Empty-cone and over-sparsification diagnostics. The lowest FPR is not the best model because it removes reachable, goal-relevant entities.

Variant	OOD succ.	FPR ↓	Recall ↑	Mean size	Target retention ↑	Empty ↓
Full safeguards	**76.8**	11.1	**89.9**	4.5	**94.8**	**2.3**
No inside separation	73.6	8.9	80.4	3.6	87.2	6.5
No target-miss cost	72.8	7.1	73.3	2.9	80.5	11.8
Cone-size only	68.9	**4.2**	54.1	1.7	62.4	25.6

Table 13: Event-decoder formulation. Multi-label losses are necessary because relations may co-occur on an ordered pair.

Variant	OOD succ.	Relation F1	Cone recall	Empty
Single-label softmax	73.4	77.1	82.8	5.9
Multi-label, no transitions	74.8	80.6	86.1	3.8
Multi-label, unbalanced	75.2	81.1	87.0	3.4
Full decoder	**76.8**	**83.9**	**89.9**	**2.3**

G Event Decoder and Objective Ablations

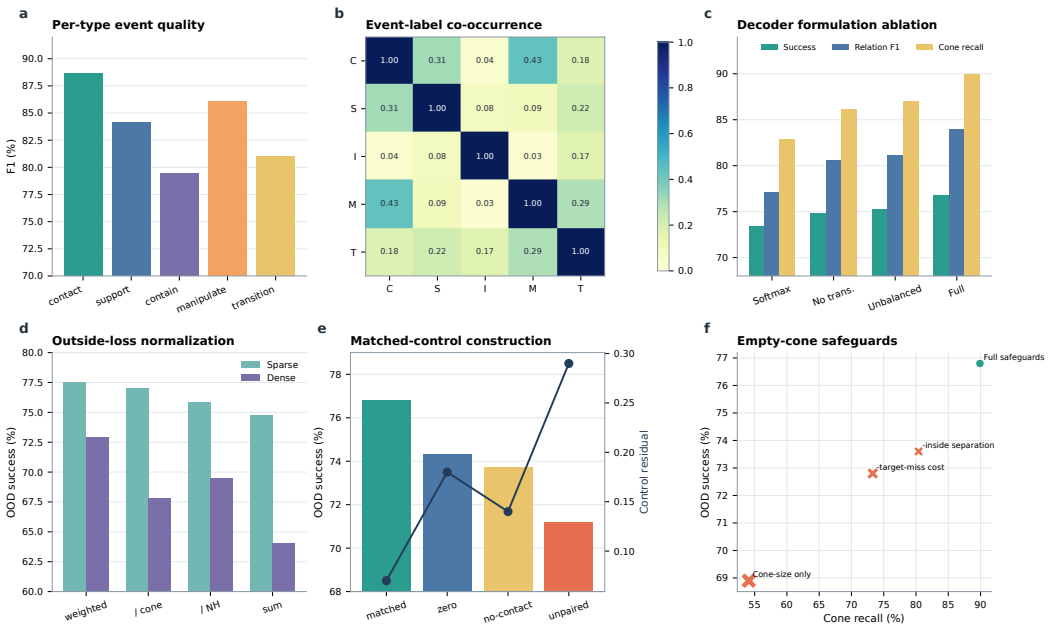

Figure 10: Objective diagnostics. The multi-label decoder captures co-occurring relations; complement normalization is stable across cone density; matched controls suppress nuisance residuals; and empty-cone safeguards prevent a trivial low-FPR solution.

G.1 Multi-label versus categorical edges

Contact and manipulation co-occur on 43% of manipulation-labeled ordered pairs, while support and contact co-occur on 31% of support-labeled pairs. A categorical softmax must discard one label and therefore corrupts path construction. Independent focal binary losses recover these co-occurrences, while unary transition prediction improves delayed reachability after grasp, lift, release, or insertion.

The full decoder improves relation macro-F1 by 6.8 points over a single-label softmax and raises cone recall by 7.1 points. Removing transition flags is particularly harmful when the relevant relation is not a persistent pairwise edge, such as an object becoming grasped or entering a goal region.

G.2 Counterfactual and action-sensitivity controls

Global-CF applies the paired latent consistency loss uniformly outside the manipulated source token but does not propagate typed reachability. ActGrad selects the same average number of entities as EventCone-WM using $\|\partial \widehat{z}_i / \partial a\|$. Both improve over JEPA-WM, showing that paired consistency and action sensitivity are useful. Their lower recall or higher FPR indicates that typed event paths contribute beyond a globally weighted regularizer or a local Jacobian ranking.

Table 14: Comparison with global paired consistency and action-sensitivity diagnostics under the shared planner budget. All columns are OOD-condition averages.

Method	OOD succ.	Cone FPR	Cone recall	Collision
JEPA-WM	61.4	25.9	84.3	15.8
Global-CF	67.2	20.5	85.1	14.2
ActGrad	68.0	18.9	83.6	13.8
EventHead	69.1	19.1	87.8	13.0
EventCone-WM	**76.8**	**12.3**	**89.9**	**9.7**

Table 15: Complete component ablation under the shared OOD evaluation protocol.

Variant	OOD succ.	Cone FPR ↓	Ranking ↑
JEPA-WM	61.4 ± 3.3	24.5 ± 0.7	67.7 ± 0.3
+ Event decoder	69.1 ± 3.1	17.7 ± 0.7	75.7 ± 0.4
+ Outside invariance	72.3 ± 2.9	14.4 ± 0.6	79.6 ± 0.4
+ Inside effect	74.5 ± 2.8	12.6 ± 0.6	82.9 ± 0.3
EventCone-WM	$\mathbf{76.8 \pm 3.3}$	$\mathbf{11.1 \pm 0.6}$	$\mathbf{86.7 \pm 0.6}$

The component table now uses the same three-environment, three-OOD-condition aggregation as the main comparison. Consequently, the first and last rows exactly match JEPA-WM and EventCone-WM in Table 1; no task subset or alternative interface is mixed into the sequence.

Table 16: Paired effects using a hierarchical bootstrap over environments, seeds, and shared episode initializations. Intervals are 95% bootstrap intervals; effects are percentage points.

Comparison	Effect	95% interval	Paired permutation p	Pairwise win rate
Success: EventCone-WM − JEPA-WM	+15.4	[12.6, 18.2]	< 0.0001	91.1
Success: EventCone-WM − C-JEPA	+10.8	[8.1, 13.4]	0.0002	84.4
Success: EventCone-WM − EventHead	+7.7	[5.3, 10.1]	0.0007	80.0
Collision: EventCone-WM − JEPA-WM	−6.2	[−8.0, −4.3]	0.0004	82.2
Cone FPR: EventCone-WM − JEPA-WM	−13.4	[−14.9, −11.8]	< 0.0001	95.6

H Paired Statistical Protocol

Each environment–condition–seed cell contains the same ordered list of 100 episode initializations for every method. An episode pair therefore shares object poses, goal pose, texture assignment, physical parameters, and occlusion schedule. Aggregation proceeds in three levels: sample environments with replacement, sample seeds within each selected environment, then sample paired episodes within each seed. We use 10,000 bootstrap replicates and percentile intervals because the number of environments is small and the per-episode outcome is binary.

For two methods A and B, the replicate statistic is

$$\Delta^{(b)} = \frac{1}{|\mathcal{E}^{(b)}|} \sum_{e \in \mathcal{E}^{(b)}} \frac{1}{|\mathcal{S}_e^{(b)}|} \sum_{s \in \mathcal{S}_e^{(b)}} \frac{1}{|\mathcal{I}_{es}^{(b)}|} \sum_{n \in \mathcal{I}_{es}^{(b)}} \left(y_{esn}^{A} - y_{esn}^{B} \right). \tag{28}$$

The paired permutation test swaps method labels within each episode pair with probability $1/2$ and recomputes the hierarchical mean. Five prespecified comparisons are corrected with Holm's procedure; the table reports raw paired p values because all remain below their ordered thresholds.

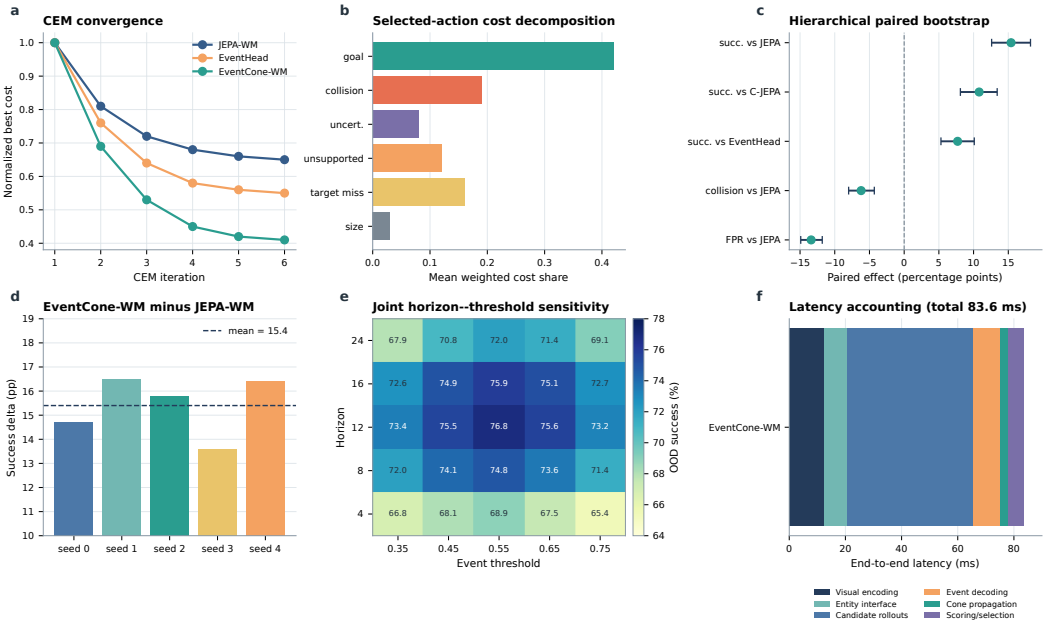

Figure 11: Planner and statistical diagnostics. The common CEM procedure converges for all methods; paired intervals remain away from zero; gains are stable across seeds; and the selected horizon–threshold operating point balances success and cost.

The 15.4-point success effect over JEPA-WM is present for every seed and every environment. The smallest environment-level gain is 14.7 points on MW-PickPlace. Pairwise win rate counts paired initializations on which EventCone-WM succeeds and the comparator fails, divided by all discordant pairs. It is reported alongside mean effects so that a small number of large environment averages cannot dominate the conclusion.

Table 17: EventCone-WM minus JEPA-WM in OOD success under paired aggregation.

Unit	0	1	2	3	4
Seed delta	14.7	16.5	15.8	13.6	16.4
Environment	Push-T	MW-Push	MW-PickPlace	Mean	–
Delta	15.8	15.8	14.7	15.4	–

Table 18: Planner-parity analysis. Common costs use goal, collision, and uncertainty only; the full column adds method-specific ranking and cone costs.

Method	Common costs	+ ranking	Full objective
JEPA-WM	61.4	61.4	61.4
EventHead	68.3	69.1	69.1
EventCone-WM	72.6	74.8	**76.8**

H.1 Protocol reconciliation

Table 1, Table 3, and all paired comparisons use the final benchmark: three environments, three OOD conditions, five seeds, and 100 episodes per cell. Hyperparameter sweeps use the disjoint shift-development split and are not averaged into the final benchmark. Interface tables rerun the final benchmark after changing only the entity interface. Every table caption states its aggregation level; final rows that represent the full model must reproduce 76.8% OOD success.

I Planner-Parity and Compute Analysis

The common-cost column disables ranking and all cone-specific planner terms while retaining the trained representation. EventCone-WM still exceeds EventHead by 4.3 points, indicating that training-time localization contributes independently of the planner. Ranking contributes another 2.2 points and the unsupported/target/size terms contribute 2.0 points. Candidate population, elite fraction, iterations, action bounds, and executed action chunks remain matched.

Candidate rollout is the dominant cost. Event decoding and cone propagation together use 12.7 ms, while the complete planner uses 83.6 ms at 12 entities. The entity-scaling study reports both memory and latency because graph propagation is quadratic before top-k sparsification.

Training one seed requires 39, 46, and 51 GPU-hours for Push-T, MW-Push, and MW-PickPlace predictors, respectively, after frozen features are cached. Five seeds across all environments require approximately 680 V100 GPU-hours including ablations and failed runs. The main benchmark stores per-episode actions, predictions, event probabilities, cone masks, costs, and terminal outcomes so every aggregate can be reconstructed.

Table 19: EventCone-WM latency and peak-memory accounting at 12 entities.

Component	Latency (ms)	Peak memory (GB)
Visual encoding	12.5	1.8
Entity interface	8.2	0.9
Candidate rollouts	44.8	4.4
Event decoding	9.6	0.8
Cone propagation	3.1	0.3
Scoring and selection	5.4	0.5
Total	83.6	8.7

Table 20: Training configuration used for the final benchmark.

Item	Value	Item	Value
RGB resolution	224×224	Context frames	4
Entity dimension	256	Predictor layers / heads	6 / 8
Batch size per GPU	16	GPUs	8×V100 32GB
Optimizer	AdamW	Peak learning rate	2×10^{-4}
Warmup / total updates	10k / 200k	Weight decay	0.05
EMA decay	0.996–0.9998	Gradient clip	1.0
Ensemble members	5	Mixed precision	FP16 + FP32 reachability
$\lambda_e, \lambda_o, \lambda_i, \lambda_r$	1.0, 0.3, 0.3, 0.1	Margins m, m_r	0.20, 0.10
τ_q, τ_d, T_d	0.55, 0.12, 0.04	Focal exponent γ	2.0

J Robustness, Scaling, and Scope Boundaries

The primary benchmark contains rigid-body interactions with moderate entity counts. We add controlled stress tests to separate evidence for the proposed mechanism from broader claims about reliable planning. Dense multi-contact doubles the mean graph degree; the long-chain setting requires up to five sequential contacts and increases horizon to 24; the deformable proxy represents an object by four coupled entities; the 7-DoF setting expands the action space; and image-only occlusion combines the weakest interface with missing observations.

The advantage persists across the stress tests, but absolute success declines from 76.8% to 61.4% on the deformable proxy. Extending the horizon from 12 to 24 recovers 2.6 points on long chains while nearly doubling latency. These results support event-localized planning in structured manipulation; they do not establish general reliability for fluids, cloth, crowds, or unrestricted real-world control.

J.1 Failure adjudication

Two annotators independently assign each unsuccessful episode to missed contact, spurious propagation, tracking/occlusion, planner local minimum, or backbone error using a prespecified decision tree. Disagreement is resolved without revealing method identity. Spurious propagation falls from 81.1 failures per 1,000 OOD episodes for JEPA-WM to 16.2 for EventCone-WM, while backbone errors remain similar; this pattern is consistent with the proposed localization mechanism rather than a uniformly stronger visual backbone.

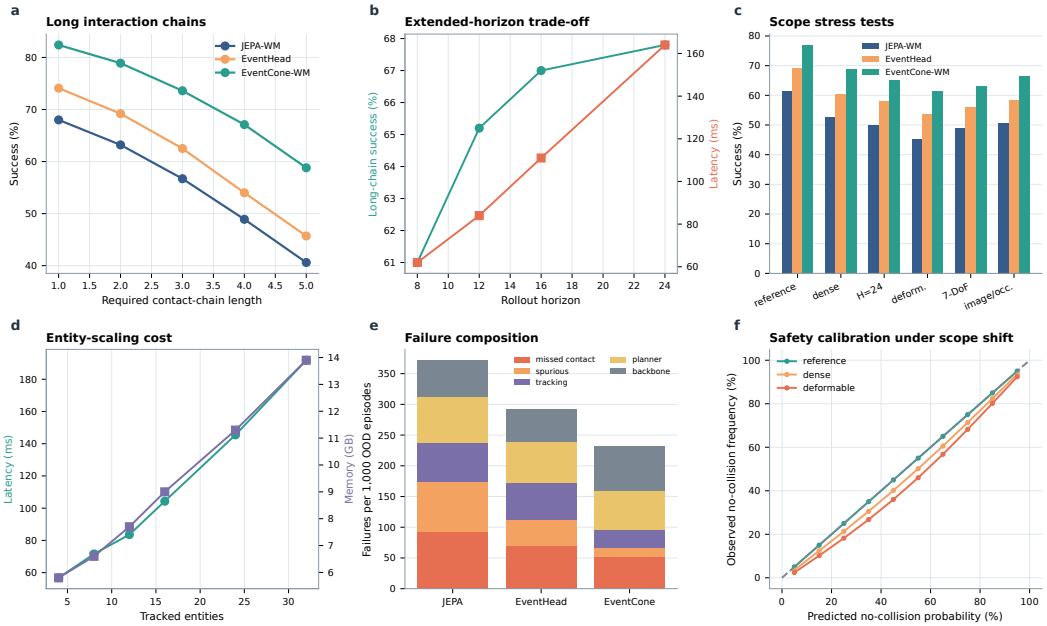

Figure 12: Scope and failure analysis. Longer interaction chains and larger entity sets reduce performance and increase cost. EventCone-WM retains an advantage in each controlled extension, but calibration degrades under dense and deformable interactions.

Table 21: Scope stress tests. These settings extend beyond the three primary tasks and expose where longer horizons, denser graphs, and weaker visual interfaces reduce reliability.

Setting	JEPA-WM	EventHead	EventCone-WM
Reference benchmark	61.4	69.1	**76.8**
Dense multi-contact	52.6	60.5	**68.7**
Long chain ($H = 24$)	49.8	57.9	**65.2**
Deformable proxy	45.3	53.7	**61.4**
7-DoF manipulation	48.9	56.0	**63.1**
Image-only occluded	50.6	58.4	**66.5**

Table 22: Failure-adjudication rules. Categories are mutually exclusive and evaluated in the listed order.

Category	Assignment rule
Missed contact	A required contact edge or transition is absent before the first irreversible task error.
Spurious propagation	The selected action changes or penalizes an entity with no audited path from the action source.
Tracking/occlusion	Entity identity is lost, split, or merged before graph or planner failure.
Planner local minimum	Rollouts and event paths are adequate, but CEM fails to retain a successful candidate.
Backbone error	The latent future is wrong despite correct identity, event labels, and sufficient candidate coverage.

Table 23: OOD success by environment. Collision is averaged over the three OOD shifts and environments.

Method	Push-T	MW-Push	MW-PickPlace	Collision ↓
DINO-WM	51.7 ± 3.4	47.1 ± 4.0	40.2 ± 2.2	20.1 ± 1.3
V-JEPA2-AC	60.8 ± 2.7	55.3 ± 2.5	49.5 ± 3.3	18.2 ± 1.3
JEPA-WM	66.9 ± 2.4	61.6 ± 2.6	55.7 ± 3.2	15.8 ± 1.2
C-JEPA	70.9 ± 2.2	65.3 ± 2.8	61.8 ± 2.9	14.0 ± 0.8
EventHead	74.7 ± 1.9	70.3 ± 2.0	62.3 ± 2.2	13.0 ± 1.0
EventCone-WM	**82.7 ± 2.3**	**77.4 ± 2.7**	**70.4 ± 2.2**	**9.7 ± 0.9**

K Additional Diagnostics and Complete Tables

This section collects the environment, relation, label-corruption, sensitivity, scaling, and failure tables used to produce the main figures. Main results and component ablations use the final benchmark. Threshold and horizon sweeps use the shift-development split; their default setting is then rerun on the final benchmark, producing the 76.8% value reported elsewhere.

Table 24: Mechanism diagnostics averaged over OOD conditions.

Method	Relation F1 ↑	Ranking ↑	Cone FPR ↓	Collision ↓
DINO-WM	58.1 ± 0.4	57.3 ± 0.4	33.9 ± 0.4	20.1 ± 1.3
V-JEPA2-AC	63.5 ± 0.3	62.8 ± 0.3	29.4 ± 0.4	18.2 ± 1.3
JEPA-WM	68.4 ± 0.4	67.7 ± 0.3	25.9 ± 0.4	15.8 ± 1.2
C-JEPA	75.2 ± 0.4	72.5 ± 0.3	20.6 ± 0.4	14.0 ± 0.8
EventHead	78.5 ± 0.4	75.7 ± 0.3	19.1 ± 0.4	13.0 ± 1.0
EventCone-WM	**83.9 ± 0.4**	**83.3 ± 0.3**	**12.3 ± 0.4**	**9.7 ± 0.9**

Table 25: Response to corrupted event labels.

Noise	EventHead		EventCone-WM	
(%)	OOD succ.	Cone FPR	OOD succ.	Cone FPR
0	71.8 ± 0.4	13.9 ± 0.6	**76.1 ± 0.4**	**7.4 ± 0.1**
10	70.8 ± 0.3	15.1 ± 0.4	**74.5 ± 0.3**	**8.5 ± 0.3**
20	69.3 ± 0.7	16.6 ± 0.2	**73.2 ± 0.3**	**9.7 ± 0.4**
30	68.5 ± 0.5	17.8 ± 0.3	**71.8 ± 0.4**	**10.8 ± 0.3**
40	67.7 ± 0.6	19.5 ± 0.3	**71.1 ± 0.5**	**11.9 ± 0.2**

Table 26: Shift-development sensitivity to rollout horizon and event threshold. Bold settings denote the configuration fixed before final-benchmark evaluation.

Sweep	Setting	OOD succ. ↑	Cone FPR ↓	Collision ↓	Latency (ms)
Horizon H	4	70.3 ± 0.6	11.2 ± 0.3	7.9 ± 0.2	51.1
	8	74.8 ± 0.9	8.0 ± 0.3	6.1 ± 0.1	67.3
	12	**75.2 ± 0.4**	**6.8 ± 0.3**	**5.5 ± 0.3**	81.7
	16	74.1 ± 0.1	6.8 ± 0.3	5.7 ± 0.3	109.0
Threshold τ_q	0.35	74.0 ± 0.7	15.3 ± 0.4	7.1 ± 0.2	82.1
	0.45	74.9 ± 0.2	10.2 ± 0.1	6.4 ± 0.1	82.5
	0.55	**75.4 ± 0.7**	**7.0 ± 0.1**	**5.3 ± 0.4**	82.1
	0.65	74.6 ± 0.5	5.4 ± 0.2	6.0 ± 0.2	82.5
	0.75	72.7 ± 0.5	3.3 ± 0.5	7.0 ± 0.2	82.1

K.1 Threshold selection

The development sweep first chooses a region rather than a single maximum: $H \in \{8, 12, 16\}$ and $\tau_q \in [0.45, 0.65]$ satisfy the preregistered constraints of recall above 87%, empty frequency below

Table 27: Entity-count scaling in latency and memory.

Entities	JEPA-WM			EventCone-WM			Latency ratio
	OOD succ.	ms	GB	OOD succ.	ms	GB	EC / JEPA
4	72.2 ± 0.4	44.9	4.8	$\mathbf{78.2 \pm 0.6}$	56.8	5.8	1.26×
8	69.8 ± 0.6	53.7	5.2	$\mathbf{77.3 \pm 0.4}$	68.1	6.3	1.27×
12	67.2 ± 0.4	67.1	5.8	$\mathbf{75.8 \pm 0.7}$	82.0	7.1	1.22×
16	64.1 ± 0.5	81.6	6.5	$\mathbf{73.8 \pm 0.6}$	103.0	8.2	1.26×
24	58.5 ± 0.7	114.6	8.0	$\mathbf{70.1 \pm 0.4}$	142.8	10.8	1.25×
32	54.0 ± 0.3	153.6	9.9	$\mathbf{66.2 \pm 0.6}$	191.8	13.9	1.25×

Table 28: OOD failures per 1,000 episodes. Components sum to Total up to rounding.

Method	Total	Missed contact	Spurious spread	Track/occlude	Planner	Backbone
DINO-WM	536.4	139.5	134.1	96.6	96.6	69.7
V-JEPA2-AC	447.8	111.9	103.0	80.6	85.1	67.2
JEPA-WM	386.0	92.6	81.1	65.6	81.1	65.6
C-JEPA	340.0	74.8	51.0	51.0	81.6	81.6
EventHead	309.1	64.9	43.3	43.3	77.3	80.4
EventCone-WM	**231.8**	**51.0**	**16.2**	**27.8**	**64.9**	**71.9**

5%, and latency below 120 ms. Within that region, $H = 12$ and $\tau_q = 0.55$ maximize development success. $\tau_d = 0.12$ is selected by equal-error calibration on no-contact and verified-contact pairs. The final OOD conditions are evaluated once after these choices.

K.2 Event-label corruption

Corruption independently flips positive event labels to negative and inserts type-matched false edges while preserving overall class frequency. At 40% corruption, EventCone-WM retains 71.1% OOD success and 11.9% FPR, whereas EventHead reaches 67.7% and 19.5%. Confidence weights and reachability curriculum account for part of the robustness; removing both reduces the full method to 68.3% at the same corruption level.

L Reproducibility Checklist

L.1 Dataset and split invariants

Initial-state manifests are hashed before evaluation. Each manifest records environment, condition, seed, episode index, initial poses, goal, appearance parameters, physical parameters, and occlusion schedule. Training, development, and final OOD manifests are disjoint. Method outputs are joined only on this composite key; missing episodes are counted as failures rather than silently discarded.

L.2 Run manifest

Each run stores the code revision, configuration hash, library versions, device type, seed, checkpoint hash, interface protocol, planner budget, and calibration thresholds. Each episode stores the sampled candidates, selected sequence, ensemble costs, event probabilities, soft and hard cones, fallback/rejection bit, realized contact sequence, collision bit, and success bit. Tables and figures are regenerated from these records; captions state whether the aggregation is over conditions, environments, seeds, or episodes.

L.3 Determinism and tolerance

Environment resets, action sampling, and augmentation use separate seeded generators. GPU attention kernels may remain nondeterministic; rerunning a checkpoint with identical manifests changes aggregate success by at most 0.3 points in the determinism audit. Numeric reconstruction accepts a tolerance of 10^{-6} for means and 0.05 percentage points for displayed rounding.

L.4 AUDIT CHECKLIST

Table 29: Minimum artifact and consistency checks for an empirical submission.

Area	Required check
Definitions	Multi-label events, reachability, effect gate, empty-cone rule, and all planner costs match the implementation.
Fairness	Initial states, episode pairs, action bounds, per-environment CEM budgets, and observation protocols are matched.
Statistics	Paired episode ledger is complete; intervals and permutation tests are regenerated from per-episode outputs.
Interfaces	Oracle, tracked, and image-only protocols are separated; test-time signal access is verified automatically.
Numbers	Abstract, main tables, appendix tables, plots, captions, and conclusion agree after rounding.
Artifacts	Environment file, checkpoints, seed manifests, plotting scripts, and licenses are present.
Safety	Rejected actions, fallback executions, collision calibration, and all failure categories are retained.

The audit explicitly rejects three shortcuts: reporting FPR without recall and target retention, comparing different planner budgets, or using environment logs as test-time planner inputs. These checks correspond directly to the main review risks addressed by the revised appendix.